Kanzlei Claus Peter PIETRUK
Heinrich-Lilienfein-Weg 5 D-76229 Karlsruhe

Europäisches Patentamt
Zweigstelle DEN HAAG
Patentlaan 2
Postbus 5818

NL-2280 HV Rijswijk

Claus Peter PIETRUK
Diplom - Physiker
Patentanwalt
European Patent & Trademark Attorney
Mandataire Agréé Près l'OEB & OHMI

Mein Zeichen / My Ref. PACT33/PCTE -es

Ihr Zeichen / Your Ref.

Datum / Date 13. AUG. 2004

Schutzrecht PCT/EP 03/08081
"Datenverarbeitungsvorrichtung und -verfahren"
Anmelder: PACT XPP Technologies AG / VORBACH / NÜCKEL / MAY

Auf den Bescheid vom 13. April 2004:

In der Anlage werden folgende Unterlagen eingereicht:

- Zeichnung(en) 3-fach.

European Patent Attorney

Anlage: wie aufgeführt

Kanzlei Claus Peter PIETRUK
Fon +49 721 46 20 34
Fax +49 721 46 93 08

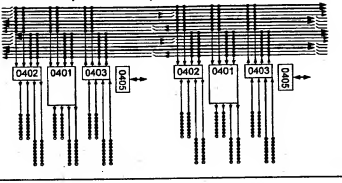
Heinrich-Lilienfein-Weg 5
USt-IDNr. DE 201 406 311 / StNr. 342 842 77 2
Baden-Württembergische Bank AG Pforzheim
Konto 4824 043 601 (BLZ 666 200 20) SWIFT BWBK DE 6S 666
Dresdner Bank AG München
Konto 4080 348 00 (BLZ 700 800 00) SWIFT DRES DE FF 700

D-76229 Karlsruhe

IN KOOPERATION MIT
RECHTSANWÄLTE
CLASSEN BAYER LAUER
KAISERSLAUTERN

PACT

- Use of (staggered) „long tracks“
  - Increases operating frequency
  - Decreases transistor count and area
  - Optimizes metalization layers
  - Decreases power dissipation




PACT

- Three additional strategies:
  - Operate at different frequencies, i.e.:
    - MIPS µP Core at 400 MHz
    - PACT XPP Core at 200 MHz
  - Registers in each bus-connect
  - PAEs act sequentially, one output each 2nd or 4th clock cycle

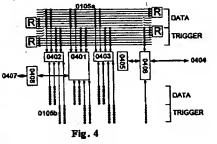



Fig. 4

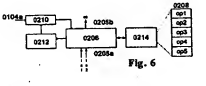



Fig. 6

PACT

## Trade-off between area and sequentiallity

- Basic operation of compiler technology can be used to achieve abstraction layer
- Generate „compressed" configurations, which are expanded on larger arrays while loading
- No minimum array but maximum array defined by compiler setting
  - Tradeoff: Number of (re-)configurations vs. usable ALU-PAEs

Fig. 8a

Fig. 8b

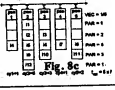


Fig. 8c

Fig. 8d





PACT

## Basic Operation Method

- LOAD/STORE processor
- RAM-PAEs act as Vector-Registers (2D/3D)
- Irregular data access patterns are linearized by LOAD/STORE while accessing RAM-PAEs
  - Can be done by µP also!
- LOAD ... $Conf_1$ ... $Conf_2$ ... ... $Conf_n$... STORE
- Each Configuration is regarded as an OpCode
- No Configuration/Array internal status

## Code Analysis

- Data Dependency Analysis
- Data Flow Analysis
- Interprocedural Alias Analysis
  - Pointer analysis: statically allocated data, dynamically allocated data
- Interprocedural Value Range Analysis

PACT

## Code Optimizations

- Loop Transformations
  - Loop Normalization
  - Loop Reversal
  - Loop-Invariant Code Motion
  - Loop Unswitching
  - Loop Interchange
  - Loop Tiling
  - Loop Skewing
  - Loop Coalescing/Collapsing
  - Loop Fusion
  - Loop Distribution
  - Loop Unrolling
  - Loop Peeling
  - Loop Splitting
  - Loop Pushing/Embedding
- Strength Reduction
- Induction Variable Elemination
- Strip Mining
- Scalar Expansion
- Array Contracting/Shrinking
- Scalar Replacement
- Reduction Recognition
- Idiom Recognition
- Procedure Inlining
- Software Pipelining
- Vector Statement Generation
- Node Splitting
- If Conversion
- Statement Reordering

XPP



PACT

- RAM-PAEs RAM is „embedded" into cache
- RAM-PAEs can operate like cache-lines
  - Homogeneous embedded in cache
  - Handling access rights between µP and XPP
  - Handling context switching / hyperthreading
  - Abstracting non linear address patterns

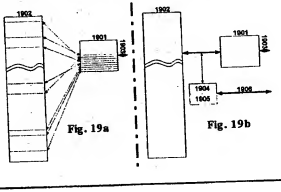



Fig. 19a

Fig. 19b

XPP

PACT33 IPC I E

4/18



PACT

- XPP operates like an RISC-Processor
- RAM-PAEs act like registers
- Each configuration is atomar (unbreakable)
- Configurations running time is limited

- LOAD Configuration
  - Loads external data into internal RAM-PAEs
- Data operations (one or multiple configurations)
  - Unbreakable – no internal status to be saved!
- STORE Configuration
  - Stores internal data into external RAM-PAEs

- Interrupts (Task/Thread-Switches) only between (re)configurations not at runtime

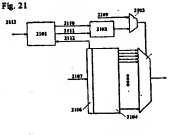
XPP



PACT

- XPP Technology allows sequential processing
  - Within ALU-PAEs using the configuration register file as a random access code memory
  - Coupling an ALU-PAE with a RAM-PAE. ALU-PAE acts like a µC, RAM-PAE is according Data- and Code-Memory
    - As an enhancement IO-PAEs can be used to access peripherials and external memory

Fig. 21



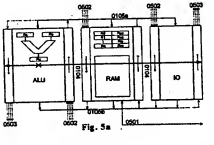



Fig. 5a

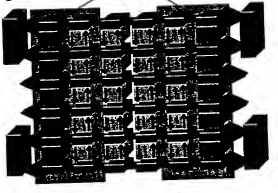
XPP

PACT

XPP

- Optimum trade-off between sequencing and dataflow processing

**Configurable ALU-PAE / RAM-PAE Sequencers**





PACT

XPP

- Handled by sequential processing within PAEs
  - I.e. Floating-Point, Division etc can be emulated by sequential multicycle PAE operations
  - Higher precision is calculated as a multicycle operation
    - results are transfered in two bus cycles

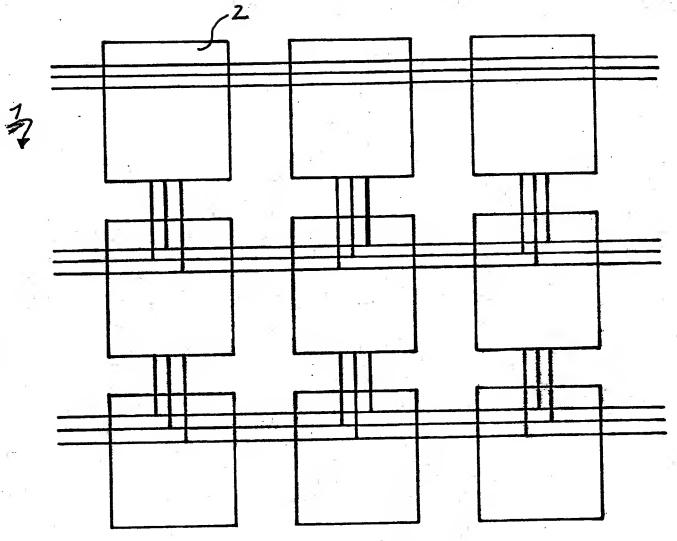



FIG. A1

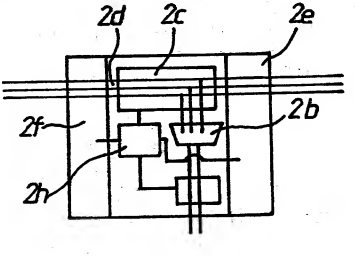



FIG. A2

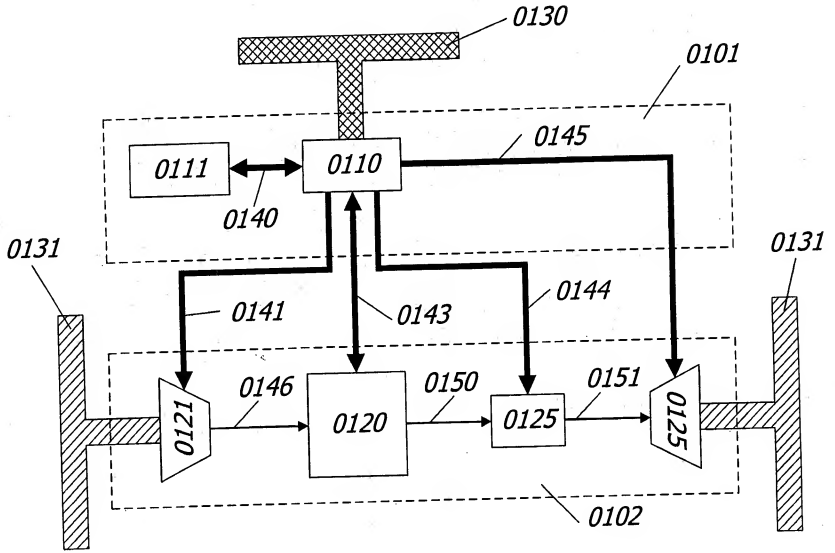

0130
0101
0111
0110
0145
0140
0131
0131
0141
0143
0144
0146
0150
0151
0121
0120
0125
0125
0102

Fig. B1

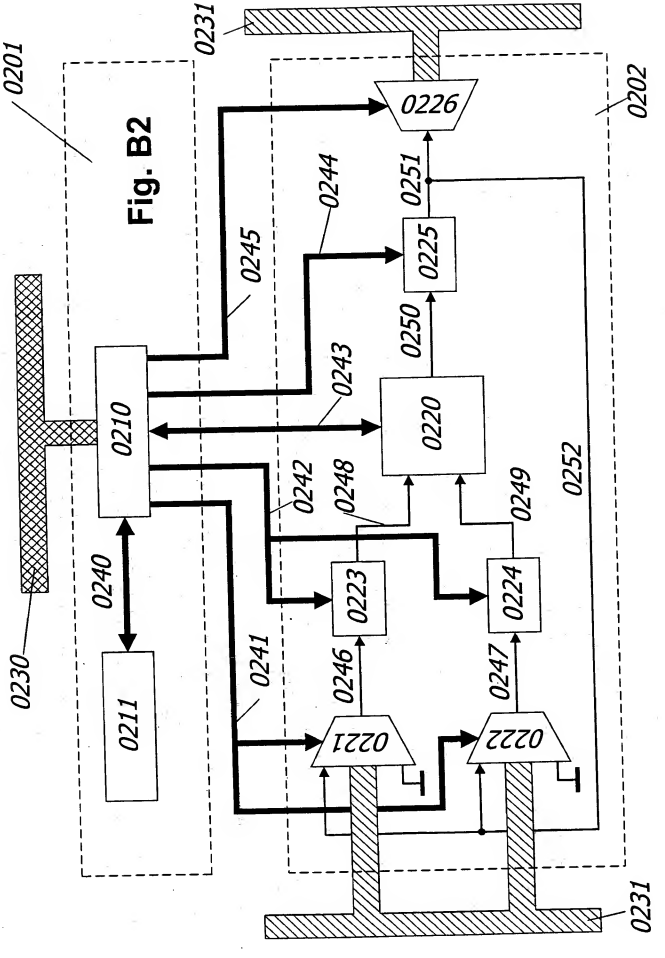
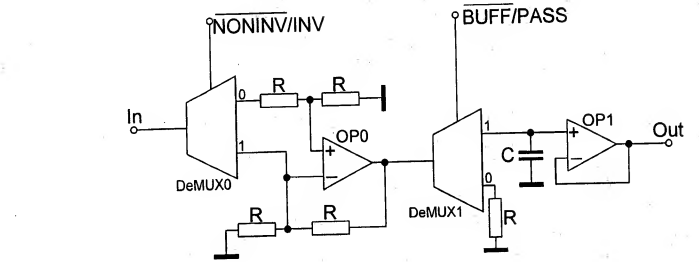

0201
0231
0202
0226
0244
0251
0225
0245
0250
0243
0210
0220
0252
0242
0248
0249
0240
0223
0224
0230
0241
0246
0247
0211
0221
0222
0231

Fig. B2


NONINV/INV
BUFF/PASS
In
0
1
R
R
OP0
DeMUX0
R
R
1
0
DeMUX1
C
OP1
Out
R

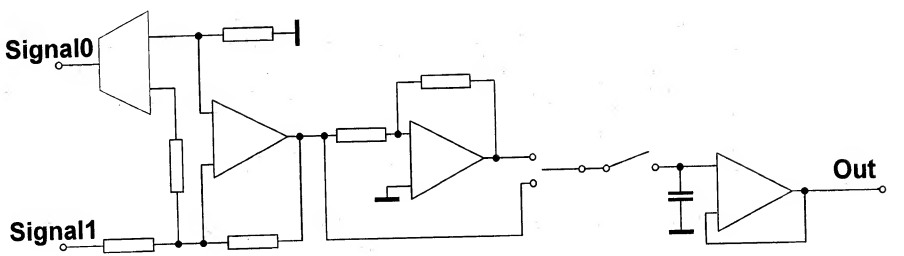

Signal0
Signal1
Out

Fig. B3

$$\{f(t)\}^{g(t)} = a^{\{g(t)\cdot\log_a(f(t))\}}$$

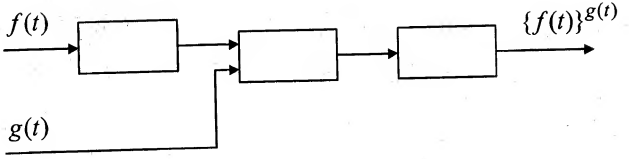



**Fig. B4**

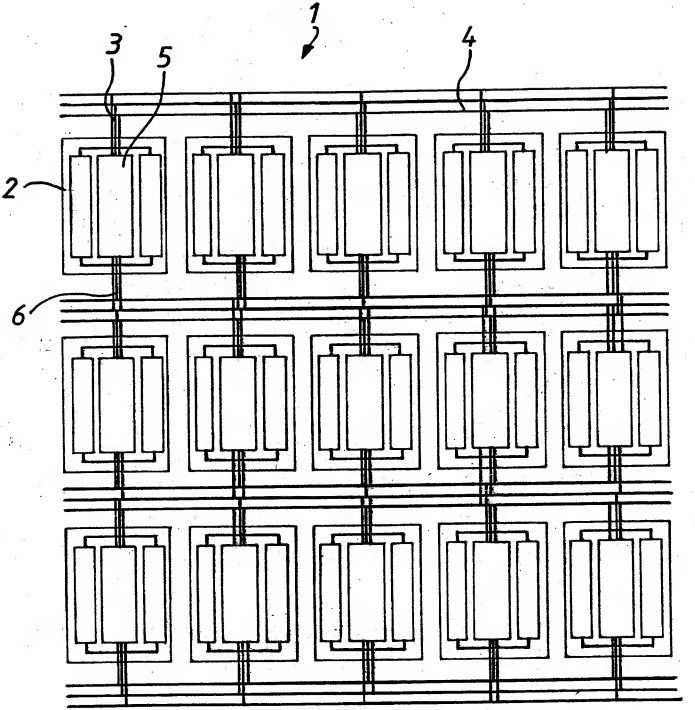



FIG. C1

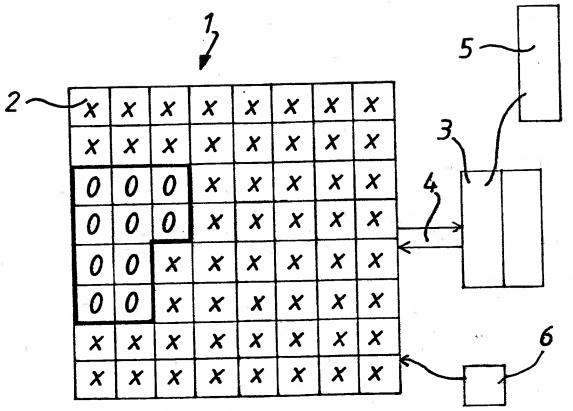



FIG. D1

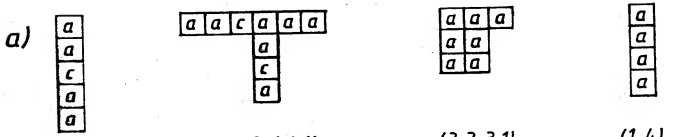



FIG. D2a

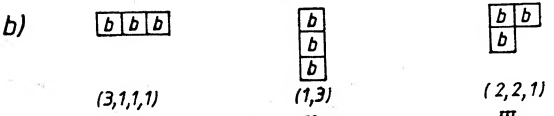



FIG. D2b

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x | x | x | x | x | x | x | x |
| x | x | x | x | x | x | x | x |
| b | b | b | x | x | x | x | x |
| a | a | a | x | x | x | x | x |
| a | a | a | x | x | x | x | x |
| a | a | x | x | x | x | x | x |
| x | x | x | x | x | x | x | x |
| x | x | x | x | x | x | x | x |

FIG. D3

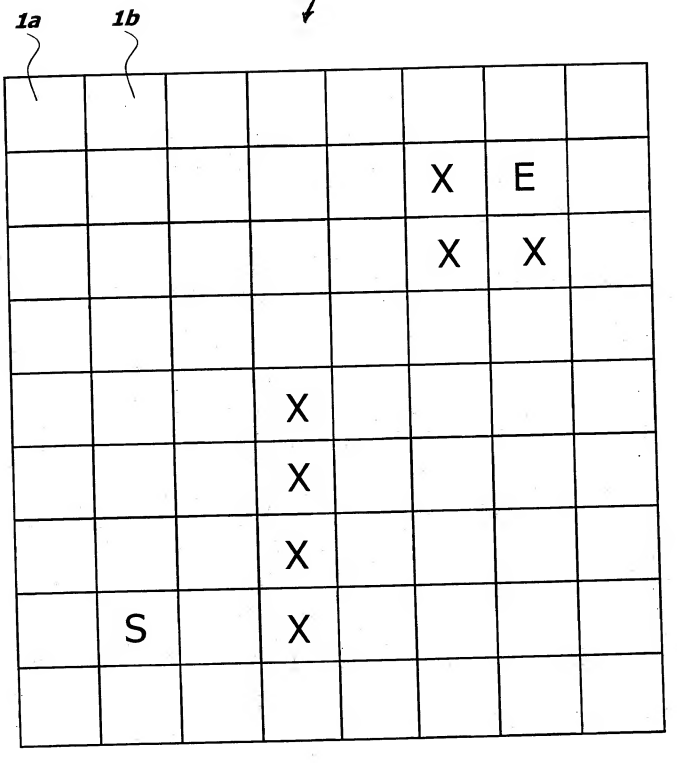



**Fig. E1**

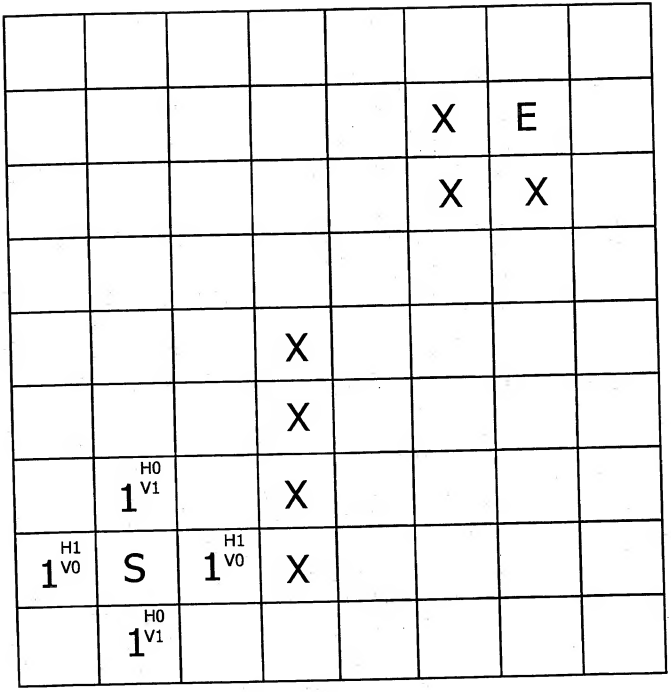



**Fig. E2**

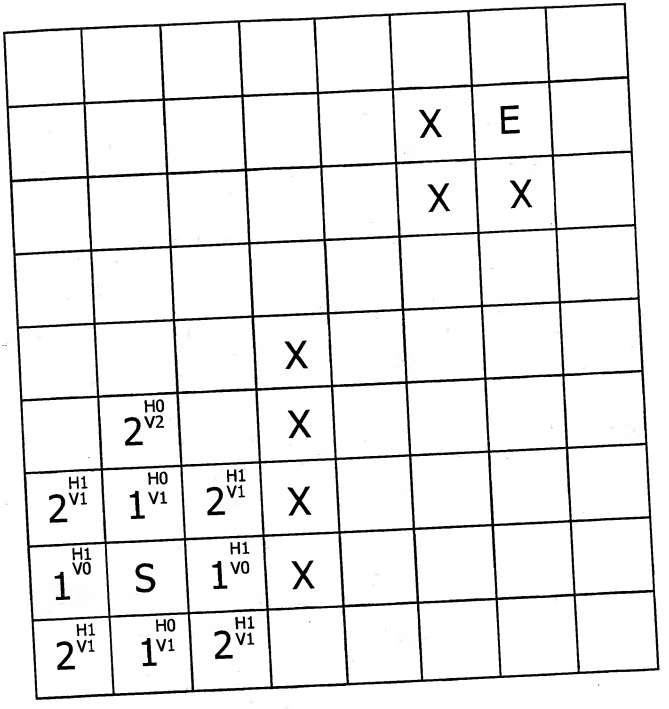

X
E
X
X
X
X
$2^{H0}_{V2}$
X
$2^{H1}_{V1}$
$1^{H0}_{V1}$
$2^{H1}_{V1}$
X
$1^{H1}_{V0}$
S
$1^{H1}_{V0}$
X
$2^{H1}_{V1}$
$1^{H0}_{V1}$
$2^{H1}_{V1}$

**Fig. E3**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| $8^{H1V7}$ | $7^{H0V7}$ | $8^{H1V7}$ | $9^{H2V7}$ | $10^{H3V7}$ | $11^{H4V7}$ | $12^{H5V7}$ | |
| $7^{H1V6}$ | $6^{H0V6}$ | $7^{H1V6}$ | $8^{H2V6}$ | $9^{H3V6}$ | X | E | $12^{H6V6}$ |
| $6^{H1V5}$ | $5^{H0V5}$ | $6^{H1V5}$ | $7^{H2V5}$ | $8^{H3V5}$ | X | X | $11^{H6V5}$ |
| $5^{H1V4}$ | $4^{H0V4}$ | $5^{H1V4}$ | $6^{H2V4}$ | $7^{H3V4}$ | $8^{H4V4}$ | $9^{H5V4}$ | $10^{H6V4}$ |
| $4^{H1V3}$ | $3^{H0V3}$ | $4^{H1V3}$ | X | $8^{H3V5}$ | $9^{H4V5}$ | $10^{H5V5}$ | $11^{H6V5}$ |
| $3^{H1V2}$ | $2^{H0V2}$ | $3^{H1V2}$ | X | $7^{H3V4}$ | $8^{H4V4}$ | $9^{H5V4}$ | $10^{H6V4}$ |
| $2^{H1V1}$ | $1^{H0V1}$ | $2^{H1V1}$ | X | $6^{H3V3}$ | $7^{H4V3}$ | $8^{H5V3}$ | $9^{H6V3}$ |
| $1^{H1V0}$ | S | $1^{H1V0}$ | X | $5^{H3V2}$ | $6^{H4V2}$ | $7^{H5V2}$ | $8^{H6V2}$ |
| $2^{H1V1}$ | $1^{H0V1}$ | $2^{H1V1}$ | $3^{H2V1}$ | $4^{H3V1}$ | $5^{H4V1}$ | $6^{H5V1}$ | $7^{H6V1}$ |

**Fig. E4**

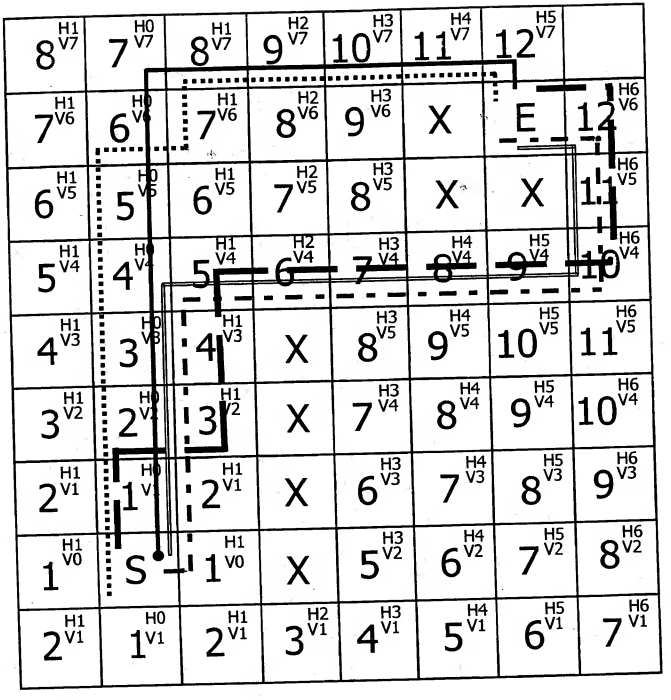

8 H1 V7
7 H0 V7
8 H1 V7
9 H2 V7
10 H3 V7
11 H4 V7
12 H5 V7
7 H1 V6
6 H0 V6
7 H1 V6
8 H2 V6
9 H3 V6
X
E
12 H6 V6
6 H1 V5
5 H0 V5
6 H1 V5
7 H2 V5
8 H3 V5
X
X
11 H6 V5
5 H1 V4
4 H0 V4
5 H1 V4
6 H2 V4
7 H3 V4
8 H4 V4
9 H5 V4
10 H6 V4
4 H1 V3
3 H0 V3
4 H1 V3
X
8 H3 V5
9 H4 V5
10 H5 V5
11 H6 V5
3 H1 V2
2 H0 V2
3 H1 V2
X
7 H3 V4
8 H4 V4
9 H5 V4
10 H6 V4
2 H1 V1
1 H0 V1
2 H1 V1
X
6 H3 V3
7 H4 V3
8 H5 V3
9 H6 V3
1 H1 V0
S
1 H1 V0
X
5 H3 V2
6 H4 V2
7 H5 V2
8 H6 V2
2 H1 V1
1 H0 V1
2 H1 V1
3 H2 V1
4 H3 V1
5 H4 V1
6 H5 V1
7 H6 V1

**Fig. E5**